

# Задание 5

## Грамматики

**Ключевые слова**<sup>1</sup>: язык, регулярный язык, ДКА, НКА, алгебра регулярных выражений, грамматики, уравнения с регулярными коэффициентами.

### 1 Грамматики

Одна из больших проблем науки, которую мы с вами изучаем – определения. Их слишком много и они отличаются друг от друга, хотя в итоге конечно описывают одни и те же классы языков. Я призываю на экзамене пользоваться определениями из книги Серебрякова, хотя при выполнении задания вы можете пользоваться эквивалентными определениями из другой литературы.

**Определение 1.** Грамматика  $\Gamma$  определяется через

- $N$  – множество нетерминальных символов
- $T$  – множество терминальных символов
- $P$  – множество правил вывода,  $P \subseteq (N \cup T)^* \times (N \cup T)^*$ .
- $S$  – аксиома,  $S \in N$ .

Все множества из описания грамматики конечные. При этом,  $N \cap T = \emptyset$ . Принято обозначение  $\Gamma = G(N, T, P, S)$ . При описании грамматики приняты следующие соглашения. Нетерминалы обозначают заглавными буквами  $A, B, C, \dots$  терминалы обозначают строчными буквами, смешанные цепочки из  $(N \cup T)^*$  обозначают греческими буквами  $\alpha, \beta, \gamma$ . Слово  $w \in T^*$  порождается грамматикой  $\Gamma$ , если существует последовательность правил вывода, начинающаяся с правила вида  $S \rightarrow \alpha$ , в результате применения которых порождается слово  $w$ . Под применением правила  $\alpha \rightarrow \beta$ , понимается, что подслово  $\alpha$  заменяется на подслово  $\beta$

---

<sup>1</sup>минимальный необходимый объем понятий и навыков по этому разделу)

В зависимости от ограничений, налагаемых на правила вывода, получаются разные классы языков. В рамках этого задания нас пока интересует только два последних типа.

- Если на множество правил  $P$  не накладывается ограничений, то есть правила имеют вид  $\alpha \rightarrow \beta$ , то грамматика называется грамматикой типа 0 по Хомскому
- Грамматики, в которых правила имеют вид  $\alpha A \beta \rightarrow \alpha \gamma \beta$ ,  $|\gamma| > 0$  называются грамматиками типа 1 или Контекстно-зависимыми. В качестве исключения грамматике может принадлежать правило  $S \rightarrow \varepsilon$ , но тогда нетерминал  $S$  не может встречаться в правых частях.
- Грамматики, в которых правила имеют вид  $A \rightarrow \alpha$ , называются грамматиками типа 2 или Контекстно-Свободными грамматиками.
- Грамматики, в которых правила имеют вид  $A \rightarrow xB$  или  $A \rightarrow x$ ,  $x \in T^*$ , называются грамматиками типа 3 или праволинейными грамматиками.

В определении КЗ-грамматики существенно, что она является *неукорачивающей*, т.е. правая часть правил всегда длиннее левой. Эквивалентное определение из книги Серебрякова гласит, что в КЗ-грамматике все правила, кроме быть может  $S \rightarrow \varepsilon$ , имеют вид  $\alpha \rightarrow \beta$ ,  $|\alpha| < |\beta|$ . Опять-таки, если есть правило  $S \rightarrow \varepsilon$ , то нетерминал  $S$  в правых частях правил встречаться не может.

Очень часто грамматиками типа 3 называют грамматики, в которых правила вывода имеют вид  $A \rightarrow xB$  или  $A \rightarrow x$ ,  $x \in T$ , также допускается правило  $S \rightarrow \varepsilon$  с всё той же оговоркой, что аксиома не может встречаться в правой части. Такие грамматики называются *праволинейными регулярными* грамматиками.

**Упражнение 1.** Доказать, что праволинейные грамматики и праволинейные регулярные грамматики эквивалентны, т.е. порождают один и тот же тип языков.

**Определение 2.** Грамматика типа 3 является *неоднозначной*, если существует более одного способа вывести хотя бы одно слово из языка, порождённого грамматикой.

Для КС-грамматик это определение неприемлемо. Вдумчивый читатель может подумать почему, прежде чем переходить к следующему разделу.

Левосторонние грамматики определяются аналогично правосторонним: в них правила имеют вид  $A \rightarrow Bx$  или  $A \rightarrow x$ .

Приведём пример КС-грамматики. При записи правил используют вспомогательное обозначение  $A \rightarrow \alpha|\beta$ , которое означает, что в грамматике есть два правила:  $A \rightarrow \alpha$  и  $A \rightarrow \beta$ .

**Пример 1.** Грамматика  $G$  задана правилами:

$$\begin{aligned} S &\rightarrow aAB \\ A &\rightarrow aA|a \\ B &\rightarrow bB|b \end{aligned}$$

Слово  $aabb$  выводится грамматикой. Последовательность применений правил вывода такая:

$$\begin{aligned} S &\rightarrow aAB \\ a\underline{A}B & \\ A &\rightarrow a \\ aa\underline{B} & \\ B &\rightarrow bB \\ aab\underline{B} & \\ B &\rightarrow b \\ aabb & \end{aligned}$$

## 1.1 Вывод, левый вывод, дерево разбора

*Выводом* цепочки  $\alpha$  называется такая последовательность применений правил с указанием раскрываемого нетерминала, что применяя правила из неё начиная с аксиомы получается цепочка  $\alpha$ . Если цепочка  $\alpha$  не содержит нетерминалов, то  $\alpha$  принадлежит языку, порождаемому КС-грамматикой. Нам будет удобно пользоваться такими понятиями как левый вывод и правый вывод. *Левым выводом* называют такой вывод, что

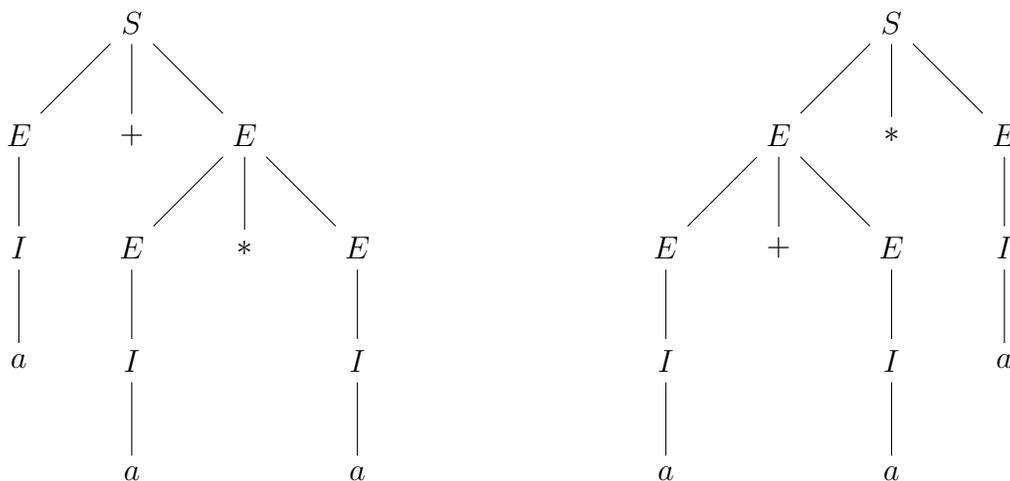
на каждом его шаге раскрывается самый левый нетерминал в промежуточной цепочке. Вывод в примере 1 является левым. Правый вывод определяется аналогично.

Также мы будем использовать деревья вывода. С формальным определением дерева вывода вы можете познакомиться, например, в книге Хопкрофта, Мотвани и Ульмана, а мы рассмотрим пример деревьев вывода и затем дадим неформальное описание этого понятия.

**Пример 2.** Грамматика  $G$ ,  $T = \{a, +, *\}$  задана правилами:

$$\begin{aligned} S &\rightarrow E \\ E &\rightarrow E + E \\ E &\rightarrow E * E \\ E &\rightarrow I \\ I &\rightarrow a \end{aligned}$$

Построим деревья вывода для слова  $a + a * a$ :



Теперь опишем понятие дерева вывода, которое также называется деревом разбора. Зафиксируем грамматику  $G$ . *Деревом вывода* для слова  $w$  называется упорядоченное дерево, в корне которого находится аксиома  $S$ , каждая вершина помечена нетерминалом, терминалом или пустым словом, если вершина помечена терминалом или  $\varepsilon$ , то эта вершина

является листом, если же вершина помечена нетерминалом  $A$ , то для некоторого правила  $A \rightarrow X_1 X_2 \dots X_n \in P$  ( $X_i \in N \cup T$ ) вершины-дети  $A$  помечены символами  $X_1, X_2 \dots X_n$  слева направо. Листья дерева вывода образуют слово  $w$ .

Как мы видим, синтаксически деревья вывода принципиально разные: в первом случае выражение интерпретируется как  $a + (a * a)$ , а во втором как  $(a + a) * a$ , что приводит к непредсказуемому результату при выполнении стандартных операций!

Эта проблема приводит нас к новому важному понятию – неоднозначности. Грамматика  $G$  является *неоднозначной*, если хотя бы для одного слова существует два различных дерева вывода.

Левый и правый вывод помимо удобства важны тем, что они фактически задают порядок обхода дерева вывода, поэтому каждому левому выводу соответствует ровно одно дерево разбора, а каждому дереву разбора соответствует ровно один левый вывод.

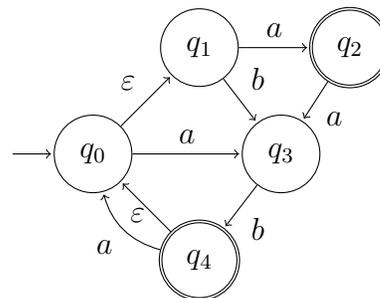
**Упражнение 2.** Доказать, что грамматика  $G$  является однозначной тогда и только тогда, когда каждое слово, порождаемое  $G$  имеет ровно один левый(правый) вывод.

## 2 Задачи

Внимание, все задачи на построение автоматов должны быть снабжены диаграммами!

### Задача 1.

Предложите алгоритм построения праволинейной грамматики по автомату  $\mathcal{A}$  и докажете его корректность. Постройте по автомату  $\mathcal{A}$  регулярную праволинейную грамматику  $G$  по предложенному алгоритму. Если вместо построения своего алгоритма, Вы возьмёте его из книжки, укажите это и не списывайте страницами, пожалуйста.



**Задача 2.** Постройте автомат по грамматике  $G$  :

$$S \rightarrow abaA|abB|\varepsilon, A \rightarrow aB|aa, B \rightarrow bA|aS$$

**Задача 3.** Является ли грамматика  $G$  из предыдущей задачи однозначной?

**Задача 4.** Язык  $L$  задан КСГ:  $S \rightarrow aSa | aSb | bSa | bSb | a$ .

1. Является ли  $L$  регулярным языком?
2. Является ли дополнение  $L$  регулярным языком?

**Задача 5.** Постройте КС-грамматику, порождающую язык

$$L = \{ a^n b^m \mid n \leq m \leq 2n \}.$$